

Driver DNP Mirror Master & Slave

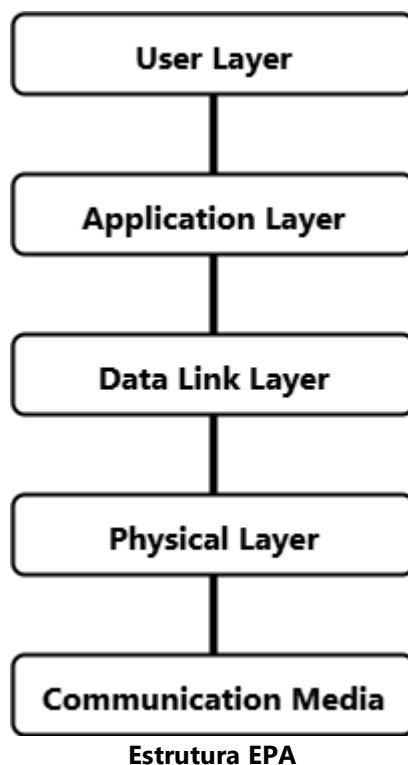
Nome do Arquivo	DNPMirror.dll
Fabricante	DNP (Distributed Network Protocol)
Equipamentos	Equipamentos compatíveis com o protocolo DNP
Protocolo	DNP 3.0
Versão	1.0.7
Última Atualização	02-09-2025
Plataforma	Win32
Dependências	IOKit versão 2.0 ou superior
Leitura com Superblocos	Não
Nível	31201

Introdução ao Protocolo DNP 3.0

O Driver DNP Mirror Master & Slave implementa o protocolo DNP versão **3.0** nos modos **Mestre** e **Escravo**, de acordo com os níveis 2 (dois) e 3 (três).

O DNP (*Distributed Network Protocol*) é um protocolo de comunicações aberto e não proprietário, baseado nas especificações da IEC (*International Electrotechnical Commission* ou *Comissão Eletrotécnica Internacional*), adaptado para ser utilizado em aplicações altamente seguras, à velocidade e quantidade de dados moderada. É extremamente flexível e pode ser utilizado em qualquer plataforma de hardware.

O modelo especificado pela ISO - OSI (*International Standards Organization - Open System Interconnection*) estabelece 7 (sete) camadas para um protocolo de rede. Já a IEC especifica um modelo simplificado, que consiste somente nas camadas **Physical** (*Física*), **Data Link** (*Dados*) e **Application** (*Aplicação*). Este modelo é chamado EPA (*Enhanced Performance Architecture*). A figura a seguir mostra a estrutura desta arquitetura e o sistema de comunicação.



A camada **User** pode ser definida como o local que o usuário manipula os dados depois de todas as comunicações. Nas aplicações da **Elipse Software**, esta camada é representada por uma aplicação de usuário. A camada **User** usa a camada **Application** deste Driver para enviar ou receber mensagens completas de e para uma estação.

A camada **Application** é responsável por especificar em detalhes os pedidos da camada **User**, e de volta àquela camada quando a mensagem vem da camada **Data Link**. Em outras palavras, esta camada junta as mensagens da camada **User**, chamadas de fragmentos, em uma mensagem de múltiplos fragmentos com informação completa para ser processada e enviada para uma estação pela camada **Data Link**.

A camada **Data Link** é usada para passar mensagens entre as estações primária ou originadora e secundária ou recebedora. Esta camada também empacota os dados, verifica erros de transmissão e envia estes dados para a rede TCP/IP.

Introdução ao Driver DNP Mirror Master & Slave

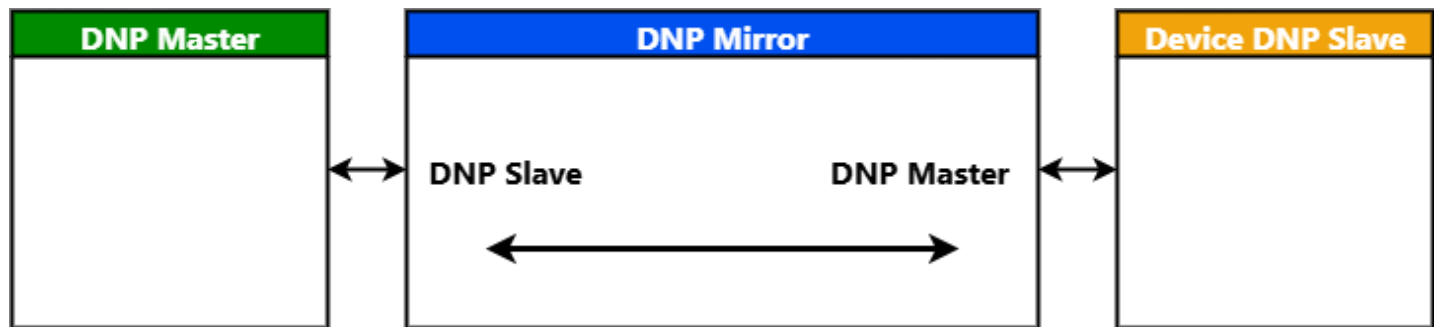
O Driver DNP Mirror Master & Slave implementa o comportamento **Mestre** e **Escravo** do protocolo DNP. O objetivo deste produto é atuar como um *gateway* DNP para DNP, adequando implementações incompatíveis deste protocolo.

Este Driver é composto de partes bem definidas, uma parte **Mestre** e uma parte **Escravo**. A parte **Mestre** deve ser configurada utilizando praticamente as mesmas opções de configuração presentes no Driver *DNP Master* da **Elipse Software**. Da mesma forma, a parte **Escravo** segue as mesmas opções de configuração existentes no Driver *DNP Slave* da **Elipse Software**.

Entretanto, neste Driver não é necessário criar nenhum Tag para que a informação recebida no lado **Mestre** seja repassada ao lado **Escravo**, pois esta operação é realizada automaticamente.

Assim, todo e qualquer valor ou evento recebido no lado **Mestre** é automaticamente repassado para o lado **Escravo**. Da mesma forma, todo comando recebido no lado **Escravo** é automaticamente repassado para o lado **Mestre**.

Pode-se dizer que o Driver DNP Mirror Master & Slave não possui uma base de pontos própria, pois obtém a base de dados a partir do **Mestre** e oferece esta mesma base para outro **Mestre** a partir de um **Escravo**.

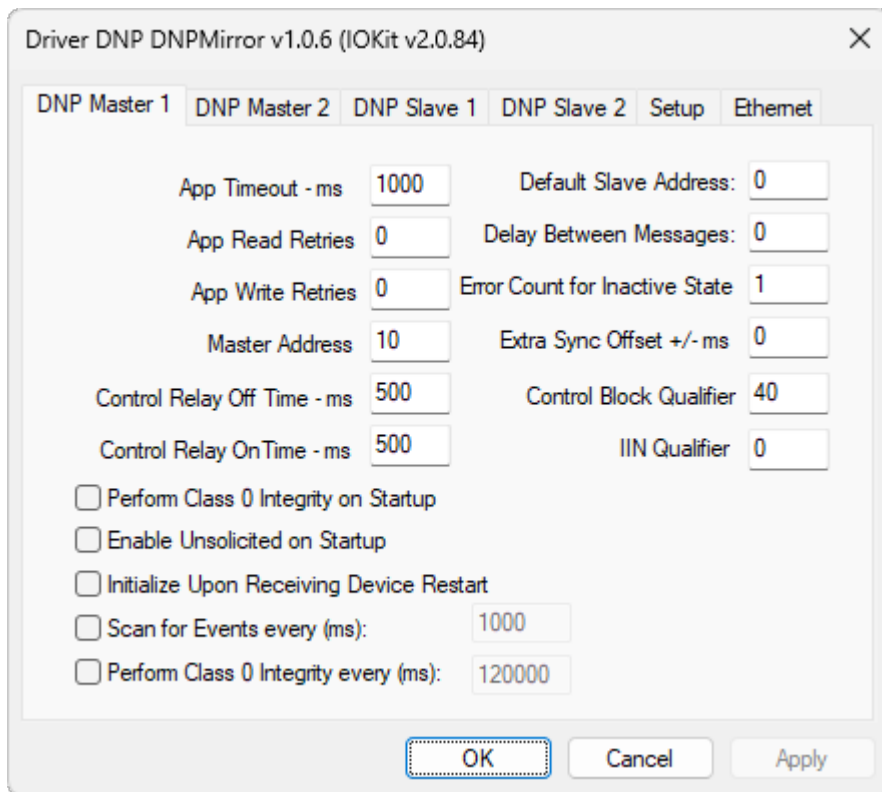


Arquitetura do Driver DNP Mirror Master & Slave

Configuração

A configuração deste Driver é realizada nas abas descritas nos tópicos a seguir.

Aba DNP Master 1



Aba DNP Master 1

As opções disponíveis nesta aba estão descritas na tabela a seguir.

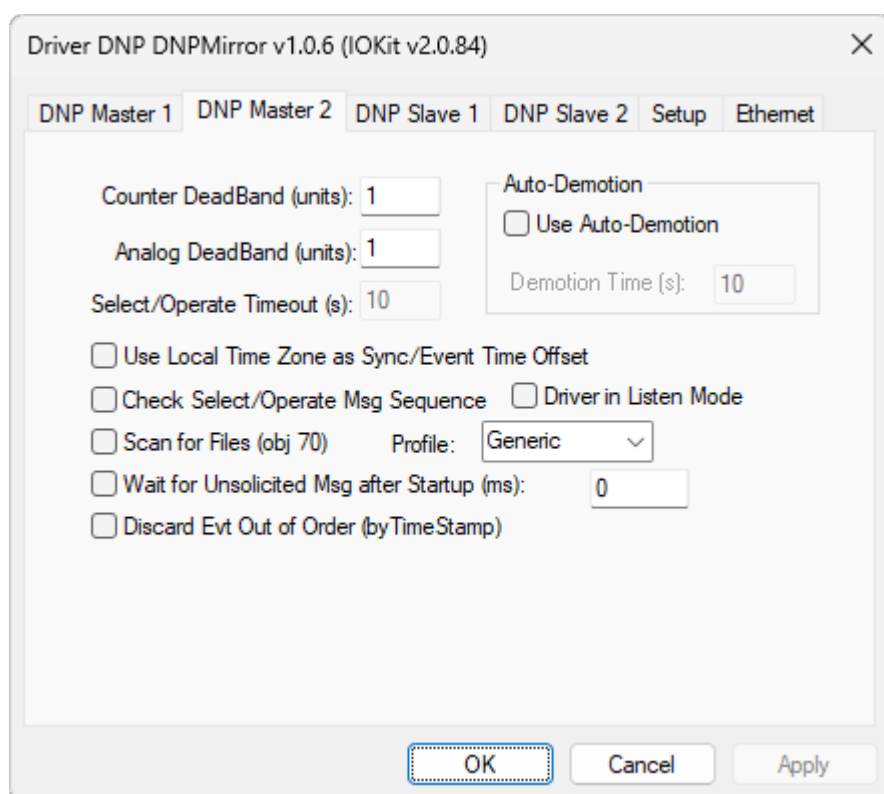
Opções disponíveis na aba DNP Master 1

OPÇÃO	DESCRIÇÃO
App Timeout - ms	Tempo máximo que a camada Application espera por uma resposta completa da camada Data Link . Se o recebimento de um pedido está em andamento pela camada Data Link , este tempo é estendido automaticamente até o término da recepção pela camada Data Link . O valor padrão desta opção depende da taxa de comunicação utilizada, mas recomenda-se que este valor seja igual ou superior ao valor de <i>time-out</i> da biblioteca IOKit , definido na aba Setup . Este valor de <i>time-out</i> da biblioteca IOKit representa o <i>time-out</i> byte a byte de uma mensagem, enquanto o valor desta opção representa uma ou mais mensagens completas da camada Data Link
App Read Retries	Número de retentativas de comunicação realizadas pela camada Application no caso de um erro de leitura. O valor padrão desta opção é 0 (zero)
App Write Retries	Número de retentativas de comunicação realizadas pela camada Application no caso de um erro de escrita. O valor padrão desta opção é 0 (zero)
Master Address	Indica o endereço da estação Mestre
Control Relay Off Time - ms	Quando são enviados comandos de relés (<i>Control Relay Commands</i>) com Tags individuais, esta opção indica o

OPÇÃO	DESCRIÇÃO
	tempo normal de Desligado ou <i>off-time</i> para os comandos Pulso On/Off ou Latch On/Off
Control Relay On Time - ms	Quando são enviados comandos de relés (<i>Control Relay Commands</i>) com Tags individuais, esta opção indica o tempo normal de Ligado ou <i>on-time</i> para os comandos Pulso On/Off ou Latch On/Off
Default Slave Address	Endereço DNP padrão de um equipamento, utilizado quando o parâmetro <i>N1</i> de cada Tag está configurado em 0 (zero)
Delay Between Messages	Tempo de atraso aplicado entre cada mensagem enviada por este Driver, em milissegundos
Error Count for Inactive State	Indica quantos erros consecutivos este Driver deve considerar para colocar um equipamento em estado Inativo . Este Driver tenta comunicar novamente com este equipamento no próximo <i>scan</i> de qualquer Tag do mesmo endereço ou somente no tempo informado na opção Demotion Time , caso este recurso seja utilizado
Extra Sync Offset +/- ms	Tempo, positivo ou negativo, adicionado aos comandos de sincronização
Control Block Qualifier	Ao realizar um comando de escrita de bloco de controle ou comando de saída digital, tais como Direct Operate , Select ou Operate , entre outros, deve-se enviar um qualificador que pode variar de equipamento para equipamento. Os valores possíveis nesta opção são 39 , ou 27h, para o campo <i>range</i> com 1 (um) byte e objeto prefixado com 2 (dois) bytes, ou 40 , ou 28h, para o campo <i>range</i> com 2 (dois) bytes e objeto prefixado com 2 (dois) bytes. Consulte o <i>device profile</i> de um equipamento para verificar o qualificador correto
IIN Qualifier	Quando um equipamento Escravo informa uma reinicialização, este Driver realiza uma escrita para o objeto de indicação interna, ou <i>Internal Indications Object</i> , informando o reconhecimento de um evento. Use esta opção para informar o qualificador, presente no documento Slave Device Profile de um equipamento Escravo . Os valores mais comuns são 0 : 1 (um) byte de índice de início e fim ou 1 : 2 (dois) bytes de índice de início e fim
Perform Class 0 Integrity on Startup	Realiza um pedido de Classe 0 ao iniciar este Driver, que obtém toda a base de dados de um equipamento para que todos os Tags sejam atualizados. Os Tags de eventos, neste caso, têm a propriedade TimeStamp como o horário da recepção dos dados e a propriedade Quality é igual a 216
Enable Unsolicited on Startup	Indica se este Driver deve enviar um comando de habilitação de mensagens não solicitadas ao iniciar a comunicação
Initialize Upon Receiving Device Restart	Indica se este Driver deve enviar os comandos de inicialização de um equipamento (Reset Link , Classe 0 e Enable Unsolicited , caso habilitados) quando receber uma mensagem de que um equipamento foi reiniciado

OPÇÃO	DESCRIÇÃO
Scan for Events every (ms)	Envia um comando de leitura das Classes 1 (um), 2 (dois) ou 3 (três) para verificar se existem eventos em um equipamento. Indique o intervalo de tempo que um comando é repetido
Perform Class 0 Integrity every (ms)	Realiza um pedido de Classe 0 ciclicamente, com o objetivo apenas de verificar se a base de dados deste Driver é igual à base de dados de um equipamento. O tempo padrão é de 10 ou 15 minutos

Aba DNP Master 2



Aba DNP Master 2

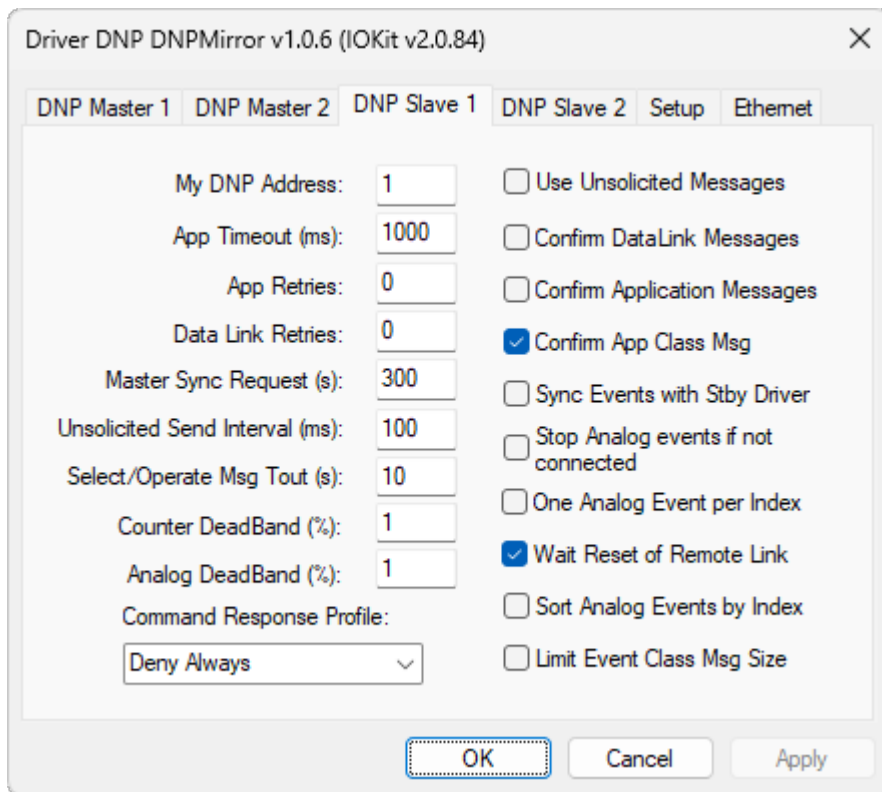
As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba DNP Master 2

OPÇÃO	DESCRIÇÃO
Counter DeadBand (units)	Banda morta, em unidades, durante a verificação de novos eventos para os contadores
Analog DeadBand (units)	Banda morta, em unidades, durante a verificação de novos eventos para os pontos analógicos
Select/Operate Timeout (s)	Tempo máximo para o bloqueio de outras mensagens, exceto a mensagem Operate , logo após um comando Select
Use Auto-Demotion	Habilita o sistema de <i>auto-demotion</i> , que retira e insere automaticamente da comunicação os Escravos que estão no estado Inativo , ou seja, com erro de comunicação. Este procedimento é utilizado quando dois ou mais Escravos

OPÇÃO	DESCRIÇÃO
	utilizam o mesmo canal, evitando assim que um Escravo neste estado monopolize a utilização de um canal
Demotion Time (s)	Tempo, em segundos, que este Driver tenta comunicar com cada Escravo no estado Inativo , para verificar se a comunicação está ocorrendo normalmente, colocando este Escravo de volta no estado Ativo
Use Local Time Zone as Sync/Event Time Offset	Instrui este Driver a considerar a diferença de horário oficial do Windows em relação ao horário em UTC (<i>Coordinated Universal Time</i>) ou GMT (<i>Greenwich Mean Time</i>) para todos os eventos e solicitações de sincronização de horário. Esta função é útil quando o lado Servidor DNP tem o horário em UTC mas é preciso informar o horário relativo local para uma aplicação
Check Select/Operate Msg Sequence	Permite a utilização de comandos Operate controlando o número de sequência de uma mensagem para que seja imediatamente subsequente ao comando Select . Ao selecionar esta opção, este Driver não envia nenhum comando, exceto o comando Operate , até que um tempo máximo para uma operação seja excedido
Driver in Listen Mode	Neste modo, este Driver não transmite nenhum dado, mas pode interpretar respostas recebidas de um <i>sniffer</i> da comunicação entre outros Mestres e um Escravo
Scan for Files (obj 70)	Indica que este Driver deve tentar realizar transferências de arquivos de um equipamento. Deve ser selecionado um perfil de coleta. Atualmente, só o perfil Pextron é suportado
Profile	Perfil de coleta. As opções disponíveis são Generic ou Pextron
Wait for Unsolicited Msg after Startup (ms)	Indica quanto tempo este Driver deve esperar por mensagens não solicitadas após conectar, com o objetivo de obter eventos recentes antes de realizar um pedido de integridade, ou Classe 0 (zero). Desta forma, os pontos que têm eventos recebem a qualidade 192 e a estampa de tempo correta, já que na resposta de integridade os objetos não possuem estampa de tempo
Discard Evt Out of Order (by TimeStamp)	Para os objetos que possuem informação de estampa de tempo, esta opção permite que eventos cuja estampa de tempo seja mais antiga que o último evento processado sejam descartados

Aba DNP Slave 1



Aba DNP Slave 1

As opções disponíveis nesta aba estão descritas na tabela a seguir.

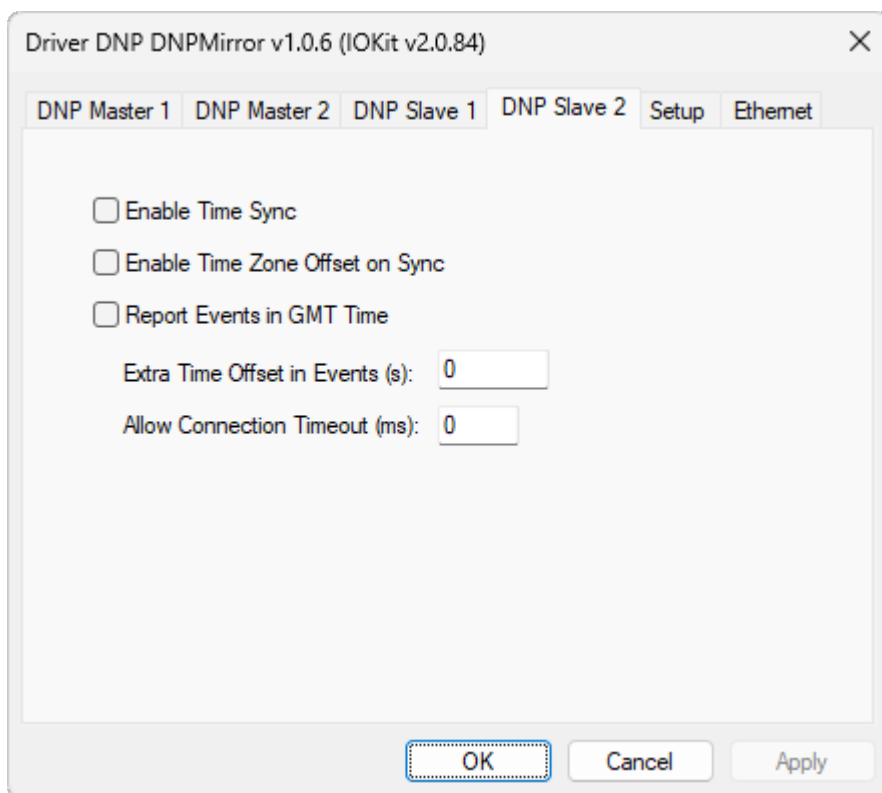
Opções disponíveis na aba DNP Slave 1

OPÇÃO	DESCRIÇÃO
My DNP Address	Endereço deste Driver (Escravo). Se há outros Escravos no mesmo <i>link</i> , este valor deve ser único
App Timeout (ms)	Tempo máximo que a camada Application espera por uma resposta completa da camada Data Link . Se o recebimento de um pedido está em andamento pela camada Data Link , este tempo é estendido automaticamente até o término da recepção pela camada Data Link . O valor padrão desta opção depende da taxa de comunicação utilizada. O <i>time-out</i> da camada Data Link , ou byte a byte, é definido pelo <i>time-out</i> da biblioteca IOKit , definido na aba Setup . O valor desta opção deve ser igual ou maior que o <i>time-out</i> da camada Data Link
App Retries	Número de retentativas de comunicação realizada pela camada Application em caso de erro de transação. O valor padrão desta opção é 0 (zero)
Data Link Retries	Número de retentativas de comunicação realizadas pela camada Data Link em caso de erro de transação. O valor padrão desta opção é 0 (zero)
Master Sync Request (s)	Intervalo de tempo que este Driver solicita a um Mestre uma sincronização de relógio, que pode ou não ser aceito, de acordo com a opção Enable Time Sync . Para desabilitar este envio, configure esta opção com o valor 0 (zero)

OPÇÃO	DESCRIÇÃO
Unsolicited Send Interval (ms)	Define o intervalo de tempo em que este Driver verifica a existência de eventos disponíveis das Classes 1 (um), 2 (dois) ou 3 (três) para serem enviados através de mensagens não solicitadas, se a opção Use Unsolicited Messages está habilitada. O lado Mestre também pode habilitar ou desabilitar o envio de mensagens não solicitadas por parte de um Escravo ao enviar as solicitações Enable e Disable Unsolicited Message , ou seja, as funções 20 e 21
Select/Operate Msg Tout (s)	Tempo máximo, em segundos, entre um comando Select e um comando Operate . Após este tempo, o comando Operate não é mais aceito por este Driver
Counter DeadBand (%)	Informa a banda morta, em porcentagem, para efeito de notificação de eventos para os contadores
Analog DeadBand (%)	Informa a banda morta, em porcentagem, para efeito de notificação de eventos para os pontos analógicos
Command Responsive Profile	Informa o tipo de tratamento que é dado aos comandos solicitados a este Driver, ou seja, os objetos 12 variação 1 (um) e 41 variações 1 (um), 2 (dois), 3 (três) e 4 (quatro). As opções disponíveis são Deny Always : Todos os comandos são respondidos negativa e instantaneamente, com o campo Status de um comando reportando o código 4 (quatro, comando não suportado), Accept Always : Todos os comandos são respondidos de forma positiva, com o campo Status de um comando reportando o código 0 (zero, comando OK) após o respectivo Tag de comando ser lido por uma aplicação. Serve para indicar que o comando solicitado foi entendido e vai ser processado por uma aplicação ou Wait for Application Response : Os comandos só são respondidos depois que o respectivo Tag é lido e escrito por uma aplicação. A leitura serve para que uma aplicação entenda que houve uma solicitação por parte do Mestre , e que é repassada para outro Driver ou saída do sistema. Após este processamento, uma aplicação deve escrever de volta no Tag o mesmo valor lido para indicar que o processamento foi bem-sucedido, ou um valor diferente caso não tenha obtido sucesso
Use Unsolicited Messages	Informa se este Driver envia mensagens não solicitadas
Confirm DataLink Messages	Informa se este Driver solicita mensagens de confirmação na camada Data Link
Confirm Application Messages	Informa se este Driver solicita mensagens de confirmação na camada Application
Confirm App Class Msg	Informa se este Driver solicita mensagens de confirmação para mensagens de classes ou eventos. NOTA : Esta opção, por segurança, é sempre habilitada internamente
Sync Events with Stby Driver	Caso exista um segundo Driver DNP Escravo em uma mesma aplicação e que esteja operando de forma redundante, ou seja, o lado Mestre seleciona em qual canal a comunicação é realizada, esta opção instrui este Driver para que, a cada mensagem de confirmação de

OPÇÃO	DESCRIÇÃO
	eventos de classes recebida, gere uma mensagem correspondente para ser enviada ao Driver redundante. NOTA: Esta mesma opção deve estar configurada no segundo Driver, a opção Confirm App Class Msg deve estar configurada em ambos os Drivers e é necessário um código auxiliar, ou script, para enviar uma mensagem de um Driver para outro
Stop Analog events if not connected	Para de acumular eventos analógicos quando a conexão com um Mestre é interrompida. O valor corrente de cada Tag analógico continua sendo armazenado. Aplica-se aos objetos 32 e 33
Wait Reset of Remote Link	Instrui este Driver a só responder comunicações após ter recebido um comando Reset of Remote Link vindo de um Mestre . O valor padrão desta opção é selecionada
Sort Analog Events by Index	Instrui este Driver a ordenar eventos analógicos, ou seja, objetos 3X com qualquer variação, pelo índice e não pela estampa de tempo, conforme é padrão para os demais eventos. Isto permite que eventos do tipo Sag/Swell de um ponto sejam reportados em conjunto, tornando possível para o lado Mestre identificar que se trata de um evento deste tipo
Limit Event Class Msg Size	Limita as mensagens de eventos a um fragmento da camada Data Link por vez

Aba DNP Slave 2



Aba DNP Slave 2

As opções disponíveis nesta aba estão descritas na tabela a seguir.

Opções disponíveis na aba DNP Slave 2

OPÇÃO	DESCRIÇÃO
Enable Time Sync	Informa se este Driver aceita pedidos de sincronização de relógio, que por sua vez sincronizam o relógio do Windows
Enable Time Zone Offset on Sync	Informa se, no momento da sincronização, este Driver considera o <i>offset</i> do horário em relação ao horário em GMT
Report Events in GMT Time	Modifica a estampa de tempo dos Tags, transformando-os para o horário GMT com base nas configurações regionais e horário de verão do Windows

Objetos Suportados

A tabela a seguir descreve todos os objetos suportados pelo Driver DNP Mirror Master & Slave.

Objetos suportados pelo Driver DNP Mirror Master & Slave

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
1	Estático	1 (um)	Binary Input without Status	1 (um)
1	Estático	2 (dois)	Binary Input with Status	1 (um)
2	Evento	1 (um)	Binary Input Change without Time	1 (um)
2	Evento	2 (dois)	Binary Input Change with Time	1 (um)
2	Evento	3 (três)	Binary Input Change with Relative Time	1 (um)
3	Estático	1 (um)	Double bit binary Input without Status	1 (um)
3	Estático	2 (dois)	Double bit binary Input with Status	1 (um)
4	Evento	1 (um)	Double bit binary Input Change without Time	1 (um)
4	Evento	2 (dois)	Double bit binary Input Change with Time	1 (um)
10	Estático	1 (um)	Binary Output	1 (um)
10	Estático	2 (dois)	Binary Output Status	1 (um)
11	Evento	1 (um)	Binary Output Event without Time	1 (um)
11	Evento	2 (dois)	Binary Output Event with Time	1 (um)
12	--	1 (um)	Control Relay Output Block	3 (três), 4 (quatro), 5 (cinco) e 6 (seis)

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
20	Estático	1 (um)	32-bit Counter	1 (um)
20	Estático	2 (dois)	16-bit Binary Counter	1 (um)
20	Estático	3 (três)	32-bit Delta Counter	1 (um)
20	Estático	4 (quatro)	16-bit Delta Counter	1 (um)
20	Estático	5 (cinco)	32-bit Counter without Flag	1 (um)
20	Estático	6 (seis)	16-bit Counter without Flag	1 (um)
20	Estático	7 (sete)	32-bit Delta Counter without Flag	1 (um)
20	Estático	8 (oito)	16-bit Delta Counter without Flag	1 (um)
21	Estático	1 (um)	32-bit Frozen Counter	1 (um)
21	Estático	2 (dois)	16-bit Frozen Counter	1 (um)
21	Estático	3 (três)	32-bit Frozen Delta Counter	1 (um)
21	Estático	4 (quatro)	16-bit Frozen Delta Counter	1 (um)
21	Estático	5 (cinco)	32-bit Frozen Counter with Time Of Freeze	1 (um)
21	Estático	6 (seis)	16-bit Frozen Counter with Time Of Freeze	1 (um)
21	Estático	7 (sete)	32-bit Frozen Delta Counter with Time Of Freeze	1 (um)
21	Estático	8 (oito)	16-bit Frozen Delta Counter with Time Of Freeze	1 (um)
21	Estático	9 (nove)	32-bit Frozen Counter without Flag	1 (um)
21	Estático	10	16-bit Frozen Counter without Flag	1 (um)
21	Estático	11	32-bit Frozen Delta Counter without Flag	1 (um)
21	Estático	12	16-bit Frozen Delta Counter without Flag	1 (um)
22	Evento	1 (um)	32-bit Counter Change Event without Time	1 (um)
22	Evento	2 (dois)	16-bit Counter Change Event without Time	1 (um)
22	Evento	3 (três)	32-bit Delta Counter Change Event without Time	1 (um)

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
22	Evento	4 (quatro)	16-bit Delta Counter Change Event without Time	1 (um)
22	Evento	5 (cinco)	32-bit Counter Change Event with Time	1 (um)
22	Evento	6 (seis)	16-bit Counter Change Event with Time	1 (um)
22	Evento	7 (sete)	32-bit Delta Counter Change Event with Time	1 (um)
22	Evento	8 (oito)	16-bit Delta Counter Change Event with Time	1 (um)
23	Evento	1 (um)	32-bit Counter Change Event without Time	1 (um)
23	Evento	2 (dois)	16-bit Frozen Counter Event without Time	1 (um)
23	Evento	3 (três)	32-bit Frozen Delta Counter Event without Time	1 (um)
23	Evento	4 (quatro)	16-bit Frozen Delta Counter without Time	1 (um)
23	Evento	5 (cinco)	32-bit Frozen Counter Event with Time	1 (um)
23	Evento	6 (seis)	16-bit Frozen Counter Event with Time	1 (um)
23	Evento	7 (sete)	32-bit Frozen Delta Counter Event with Time	1 (um)
23	Evento	8 (oito)	16-bit Frozen Delta Counter Event with Time	1 (um)
30	Estático	1 (um)	32-bit Analog Input	1 (um)
30	Estático	2 (dois)	16-bit Analog Input	1 (um)
30	Estático	3 (três)	32-bit Analog Input without Flag	1 (um)
30	Estático	4 (quatro)	16-bit Analog Input without Flag	1 (um)
30	Estático	5 (cinco)	32-bit Analog Input Floating Point	1 (um)
31	Estático	1 (um)	32-bit Frozen Analog Input	1 (um)
31	Estático	2 (dois)	16-bit Frozen Analog Input	1 (um)

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
31	Estático	3 (três)	32-bit Frozen Analog Input with Time Of Freeze	1 (um)
31	Estático	4 (quatro)	16-bit Frozen Analog Input with Time Of Freeze	1 (um)
31	Estático	5 (cinco)	32-bit Frozen Analog Input without Flag	1 (um)
31	Estático	6 (seis)	16-bit Frozen Analog Input without Flag	1 (um)
31	Estático	7 (sete)	32-bit Frozen Analog Input Floating Point	1 (um)
32	Evento	1 (um)	32-bit Change Event without Time	1 (um)
32	Evento	2 (dois)	16-bit Change Event without Time	1 (um)
32	Evento	3 (três)	32-bit Analog Change with Time	1 (um)
32	Evento	4 (quatro)	16-bit Analog Change Event with Time	1 (um)
32	Evento	5 (cinco)	32-bit Analog Change Floating Point without Time	1 (um)
32	Evento	7 (sete)	32-bit Analog Change Floating Point with Time	1 (um)
33	Evento	1 (um)	32-bit Frozen Analog Event without Time	1 (um)
33	Evento	2 (dois)	16-bit Frozen Analog Event without Time	1 (um)
33	Evento	3 (três)	32-bit Frozen Analog Event with Time	1 (um)
33	Evento	4 (quatro)	16-bit Frozen Analog Event with Time	1 (um)
33	Evento	5 (cinco)	32-bit Frozen Analog Floating Point without Time	1 (um)
33	Evento	7 (sete)	32-bit Frozen Analog Floating Point with Time	1 (um)
34	Estático	1 (um)	16-bit Analog Input Dead band	1 (um)
34	Estático	2 (dois)	32-bit Analog Input Dead band	1 (um)
34	Estático	3 (três)	32-bit Analog Input Floating Point Dead	1 (um)

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
			band	
40	Estático	1 (um)	32-bit Analog Output Status	1 (um)
40	Estático	2 (dois)	16-bit Analog Output Status	1 (um)
40	Estático	3 (três)	32-bit Analog Output Status Floating Point	1 (um)
41	--	1 (um)	32-bit Analog Output Block	2 (dois), 3 (três), 4 (quatro), 5 (cinco) e 6 (seis)
41	--	2 (dois)	16-bit Analog Output Block	2 (dois), 3 (três), 4 (quatro), 5 (cinco) e 6 (seis)
41	--	3 (três)	32-bit Floating Point Analog Output Block	2 (dois), 3 (três), 4 (quatro), 5 (cinco) e 6 (seis)
42	Evento	1 (um)	32-bit Analog Output Event without Time	1 (um)
42	Evento	2 (dois)	16-bit Analog Output Event without Time	1 (um)
42	Evento	3 (três)	32-bit Analog Output Event with Time	1 (um)
42	Evento	4 (quatro)	16-bit Analog Output Event with Time	1 (um)
42	Evento	5 (cinco)	32-bit Floating Point Analog Output Event without Time	1 (um)
42	Evento	7 (sete)	32-bit Floating Point Analog Output Event with Time	1 (um)
50	Estático	1 (um)	Time and Date	1 (um) e 2 (dois)
51	Estático	1 (um)	Time and Date CTO (Common Time of Occurrence)	1 (um)
51	Estático	2 (dois)	Unsynchronized Time and CTO (Common Time of Occurrence)	1 (um)
52	Estático	2 (dois)	Time Delay Fine	1 (um)
60	Estático	1 (um)	Class 0 Data	1 (um)
60	Evento	2 (dois)	Class 1 Data	1 (um)
60	Evento	3 (três)	Class 2 Data	1 (um)
60	Evento	4 (quatro)	Class 3 Data	1 (um)
80	--	1 (um)	Internal Indications	1 (um) e 2 (dois)
83	Evento	1 (um)	Pro	1 (um)

CÓDIGO DO OBJETO	TIPO	VARIAÇÃO	NOME DO OBJETO	FUNÇÃO
110	Estático	X	Octet String	1 (um)
111	Evento	X	Octet String Event	1 (um)

Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **DNPMirror**.

Configurações de um Driver

A configuração das Interfaces de Comunicação é realizada na caixa de diálogo de configuração de um Driver. Para acessar a configuração da caixa de diálogo no **Elipse E3** na versão 1.0, siga estes passos:

1. Clique com o botão direito do mouse em um objeto Driver (IODriver).
2. Selecione o item **Propriedades** no menu contextual.
3. Selecione a aba **Driver**.
4. Clique em **Outros parâmetros**.

No **Elipse E3** versão 2.0 ou posterior, clique em **Configurar o driver**  na barra de ferramentas de um Driver. No **Elipse SCADA**, siga estes passos:

1. Abra o Organizer.
2. Selecione um Driver na árvore do Organizer.
3. Clique em **Extras** na aba **Driver**.

Atualmente, as Interfaces de Comunicação permitem que apenas uma conexão seja aberta para cada Driver. Isto significa que, no caso de acesso a duas portas seriais, é preciso adicionar dois Drivers em um aplicação e configurar cada um destes Drivers para cada porta serial.

Caixa de Diálogo de Configuração

A caixa de diálogo das Interfaces de Configuração permite configurar a conexão de I/O que é utilizada por um Driver. Esta caixa de diálogo contém as abas **Setup**, **Serial**, **Ethernet**, **Modem** e **RAS** descritas nos tópicos a seguir. Se um Driver não implementa uma conexão de I/O específica, a respectiva aba não está disponível para configuração. Alguns Drivers podem conter abas adicionais, específicas para aquele Driver, na caixa de diálogo de configuração.

Aba Setup

A aba **Setup** contém a configuração geral de um Driver. Esta aba é dividida nos seguintes grupos:

- **Configurações gerais:** Configurações da camada física de um Driver, *time-out* e modo de inicialização
- **Connection management:** Configurações de como a Interface de Comunicação mantém a conexão e qual a política de recuperação em caso de falha
- **Logging options:** Controla a geração dos arquivos de log

Setup

Physical Layer: Ethernet ▾
 Start driver OFFLINE

Timeout: 1000 ms
Communication check time: 5000 ms

Connection management

Mode: Automatic (managed by the driver) ▾

Retry failed connection every 20 seconds

Give up after 1 failed retries

Disconnect if non-responsive for 0 seconds

Logging Options

Log to File: C:\eeLogs\MicrolokII_%DATE%.log

File size limit (MB): 0 ('0' is unlimited)

Aba Setup

Opções gerais da aba Setup

OPÇÃO	DESCRIÇÃO
Physical Layer	Selecione a interface física em uma lista. As opções disponíveis são Serial , Ethernet , Modem e RAS . A interface selecionada deve ser configurada na aba específica
Timeout	Configure o <i>time-out</i> , em milissegundos, para a camada física. Esta é a medida de tempo que a interface de I/O aguarda para a recepção de um byte qualquer do <i>buffer</i> de recepção
Communication check time	Configure o tempo, em milissegundos, para definir o intervalo em que a comunicação é considerada em estado inativo. Enquanto um Driver de Comunicação receber dados válidos, o estado de comunicação é considerado ativo. Porém, se durante o funcionamento um Driver de Comunicação não receber dados válidos neste período de tempo, o estado é considerado inativo. O estado de comunicação é mostrado no Tag IO.CommunicationStatus
Start driver OFFLINE	Selecione esta opção para que um Driver inicie em modo Offline ou parado. Isto significa que a interface de I/O não é criada até que se configure um Driver em modo Online utilizando-se um Tag em uma aplicação. Este modo possibilita a configuração dinâmica da interface de I/O em tempo de execução

Opções para o grupo Connection management

OPÇÃO	DESCRIÇÃO
Mode	Selecione o modo de gerenciamento de conexão. Selecionar a opção Automatic permite que um Driver gerencie a conexão automaticamente, como especificado nas opções seguintes. Selecionar a opção Manual permite que uma aplicação gerencie a conexão completamente
Retry failed connection every ... seconds	Selecione esta opção para habilitar a retentativa de conexão de um Driver em um determinado intervalo, em segundos. Se a opção Give up after failed retries não está selecionada, este Driver continua retentando até que a conexão seja efetuada, ou que a aplicação seja parada
Give up after ... failed retries	Habilite esta opção para definir um número máximo de retentativas de conexão. Quando o número especificado de tentativas consecutivas de reconexão é atingido, um Driver vai para o modo Offline , assumindo que um problema de hardware foi detectado. Se um Driver estabelece uma conexão com sucesso, o número de retentativas sem sucesso é zerado. Se esta nova conexão é perdida, então o contador de retentativas inicia do zero
Disconnect if non-responsive for ... seconds	Habilite esta opção para forçar um Driver a se desconectar se nenhum byte chegou à interface de I/O no <i>time-out</i> especificado, em segundos. Este <i>time-out</i> deve ser maior que o <i>time-out</i> configurado na opção Timeout

Opções para o grupo Logging Options

OPÇÃO	DESCRIÇÃO
Log to File	<p>Habilite esta opção e configure o nome do arquivo onde o log é escrito. Arquivos de log podem ser bem extensos, portanto utilize esta opção por curtos períodos de tempo, apenas para o propósito de testes e depurações. Caso se utilize a macro %PROCESS% no nome do arquivo de log, esta é substituída pelo identificador do processo atual. Esta opção é particularmente útil ao se utilizar várias instâncias de um mesmo Driver no Elipse E3, permitindo assim que cada instância gere um arquivo separado de log. Por exemplo, ao configurar esta opção com o valor "c:\e3logs\drivers\sim_%PROCESS%.log", gera-se um arquivo c:\e3logs\drivers\sim_00000FDA.log para o processo 0FDAh. Pode-se também utilizar a macro %DATE% no nome do arquivo. Neste caso é gerado um arquivo de log por dia, no formato aaaa_mm_dd. Por exemplo, ao configurar esta opção com o valor "c:\e3logs\drivers\sim_%DATE%.log", gera-se o arquivo c:\e3logs\drivers\sim_2005_12_31.log em 31/12/2005 e o arquivo c:\e3logs\drivers\sim_2006_01_01.log em 01/01/2006. De forma semelhante, a macro %DATE_HOUR% gera um arquivo de log por hora, no formato aaaa_mm_dd_hh</p>
File size limit (MB)	<p>Configure o limite de tamanho do arquivo de log, em megabytes. Um valor igual a 0 (zero) significa que não há limite de tamanho para o arquivo de log</p>

Aba Ethernet

Utilize esta aba para configurar os parâmetros da Interface **Ethernet**. Estes parâmetros, exceto as configurações de porta, devem ser também configurados para uso na Interface **RAS**.

Ethernet

Transport: TCP/IP ▼

PING before connecting
 Timeout: 4000 ms
 Retries: 1

Listen for connections on port: 0
 Share listen port with other processes
 Interface: (All Interfaces) ▼
 Use IPv6 Use SSL SSL Settings
 Enable 'ECHO' suppression
 IP Filter:

Connect to

<input type="checkbox"/> Main IP: 	Port: 502	<input type="checkbox"/> Local port: 0
<input type="checkbox"/> Backup IP 1: 	Port: 0	<input type="checkbox"/> Local port: 0
<input type="checkbox"/> Backup IP 2: 	Port: 0	<input type="checkbox"/> Local port: 0
<input type="checkbox"/> Backup IP 3: 	Port: 0	<input type="checkbox"/> Local port: 0

Aba Ethernet

Opções disponíveis na aba Ethernet

OPÇÃO	DESCRIÇÃO
Transport	Selecione o valor TCP/IP para um <i>socket</i> TCP (<i>stream</i>) ou selecione o valor UDP/IP para utilizar um <i>socket</i> UDP (<i>connectionless datagram</i>)
Listen for connections on port	Utilize esta opção para aguardar por novas conexões em uma porta IP específica, comum em Drivers Escravos. Caso esta opção permaneça desmarcada, um Driver se conecta ao endereço e porta especificados no grupo Connect to
Share listen port with other processes	Selecione esta opção para compartilhar a porta de escuta com outros Drivers e processos
Interface	Selecione a interface de rede local, identificada pelo endereço IP, que um Driver utiliza para efetuar e receber conexões, ou selecione o valor (All Interfaces) para permitir conexões em qualquer interface de rede
Use IPv6	Selecione esta opção para forçar um Driver a utilizar endereços no formato IPv6 em todas as conexões Ethernet. Deixe esta opção desmarcada para utilizar o formato IPv4
Enable 'ECHO' suppression	Habilite esta opção para eliminar o <i>eco</i> dos dados recebidos. O <i>eco</i> é uma cópia dos dados enviados, que pode ser retornada antes da mensagem de resposta
IP Filter	Lista de endereços IP restringidos ou permitidos de onde um Driver aceita conexões (<i>Firewall</i>). Consulte a propriedade IO.Ethernet.IPFilter para mais informações
PING before connecting	Habilite esta opção para executar um comando ping , ou seja, para verificar se um dispositivo pode ser encontrado na rede, em um dispositivo antes de tentar uma conexão

OPÇÃO	DESCRIÇÃO
	<p>com o <i>socket</i>. Esta é uma maneira rápida de determinar uma conexão bem sucedida antes de tentar abrir um <i>socket</i> com um dispositivo. O <i>time-out</i> de uma conexão com um <i>socket</i> pode ser bem alto. As opções disponíveis são:</p> <ul style="list-style-type: none"> • Timeout: Especifique o número de milissegundos de espera por uma resposta de um comando ping. Deve-se usar um comando ping para verificar o tempo normal de resposta, configurando esta opção para um valor acima desta média. Normalmente pode-se configurar um valor entre 1000 e 4000 milissegundos, ou seja entre 1 (um) e 4 (quatro) segundos • Retries: Número de retentativas de um comando ping, sem contar a tentativa inicial. Se todas as tentativas falharem, então a conexão com o <i>socket</i> é abortada

Opções disponíveis no grupo Connect to

OPÇÃO	DESCRIÇÃO
Main IP	Digite o endereço IP de um dispositivo remoto. Pode-se usar tanto o endereço IP separado por pontos quanto uma URL. No caso de uma URL, um Driver usa o serviço de DNS disponível para mapear a URL para um endereço IP, como por exemplo "192.168.0.13" ou "Server1"
Port	Digite a porta IP de um dispositivo remoto, entre 0 (zero) e 65535
Local port	Selecione esta opção para utilizar uma porta IP local fixa ao conectar a um dispositivo remoto
Backup IP 1, 2 e 3	Indique o endereço IP, a porta IP e a porta IP local fixa de até 3 (três) endereços de <i>backup</i> de um dispositivo remoto

Configurações Gerais

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** gerais das Interfaces de Comunicação.

Tags de Comunicação

Tags Gerais das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir são fornecidos para todas as Interfaces de I/O suportadas.

IO.CommunicationStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	6 (seis)
Configuração por String	IO.CommunicationStatus

Este Tag informa o estado da comunicação de um Driver. Indica o funcionamento da comunicação em função do recebimento de dados válidos dentro de um período de tempo arbitrado na configuração. Para mais informações, consulte o tópico **Aba Setup**. Os valores possíveis são **0 - Comunicação inativa**: O Driver não recebeu dados válidos ou deixou de receber dados depois de n milissegundos, conforme configurado na janela de propriedades, ou **1 - Comunicação ativa**: O Driver está recebendo dados válidos.

IO.IOKitEvent

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	1 (um)
Propriedade Size	4 (quatro)
Propriedade ParamItem	IO.IOKitEvent

Este Bloco retorna eventos de Driver gerados por várias fontes nas Interfaces de Comunicação. A propriedade **TimeStamp** de um Bloco representa o momento em que um evento ocorre. Os Elementos de Bloco são os seguintes:

- **Elemento 0**: Tipo de evento. Os valores possíveis são **0**: Informação, **1**: Advertência ou **2**: Erro
- **Elemento 1**: Fonte de um evento. Os valores possíveis são **0**: Driver (específico de um Driver), **-1**: IOKit (eventos genéricos da Interface de Comunicação), **-2**: Interface **Serial**, **-3**: Interface **Modem**, **-4**: Interface **Ethernet** ou **-5**: Interface **RAS**
- **Elemento 2**: Número do erro, específico de cada fonte de evento
- **Elemento 3**: Mensagem de um evento, uma **String** específica de cada evento

NOTA

Um Driver mantém um número máximo de 100 eventos internamente. Se eventos adicionais são reportados, os eventos mais antigos são descartados.

IO.PhysicalLayerStatus

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	2 (dois)
Configuração por String	IO.PhysicalLayerStatus

Este Tag indica o estado da camada física. Os valores possíveis são os seguintes:

- **0:** Camada física parada, ou seja, um Driver está em modo **Offline**, a camada física falhou ao inicializar ou excedeu o número máximo de tentativas de reconexão
- **1:** Camada física iniciada mas não conectada, ou seja, um Driver está em modo **Online**, mas a camada física não está conectada. Se a opção **Connection management** está configurada com o valor **Automatic**, a camada física pode estar conectando, desconectando ou esperando por uma tentativa de reconexão. Se a opção **Connection management** está configurada com o valor **Manual**, então a camada física permanece neste estado até ser forçada a conectar
- **2:** Camada física conectada, ou seja, a camada física está pronta para ser usada. Isto **NÃO** significa que um equipamento esteja conectado, apenas que a camada de acesso está funcionando

IO.SetConfigurationParameters

Tipo de Tag	Tag Bloco
Tipo de Acesso	Somente Leitura
Parâmetro B1	-1 (menos um)
Parâmetro B2	0 (zero)
Parâmetro B3	0 (zero)
Parâmetro B4	3 (três)
Propriedade Size	2 (dois)
Propriedade ParamItem	IO.SetConfigurationParameters

Use este Tag para modificar qualquer propriedade da caixa de diálogo de configuração de um Driver em tempo de execução.

Este Tag funciona somente enquanto um Driver está em modo **Offline**. Para iniciar um Driver em modo **Offline**, selecione a opção **Start driver OFFLINE** na caixa de diálogo de configuração deste Driver. Pode-se tanto escrever em um Tag PLC ou em um Tag Bloco contendo os parâmetros a serem modificados. As escritas de Elementos de Bloco individuais não são suportadas, um Bloco inteiro precisa ser escrito de uma vez só.

No **Elipse SCADA** é necessário usar um Tag Bloco. Cada parâmetro a ser configurado utiliza dois Elementos de Bloco. Por exemplo, caso seja necessário configurar 3 (três) parâmetros, então o tamanho do Bloco deve ser 6 (seis, 3×2). O primeiro Elemento é o nome da propriedade, como uma **String**, e o segundo Elemento é o valor desta propriedade, conforme o exemplo a seguir.

```
// 'Block' deve ser um Tag Bloco com leitura automática,
// leitura por varredura e escrita automática desabilitadas.
// Configura os parâmetros
Block.element001 = "IO.Type" // Parâmetro 1
Block.element002 = "Serial"
Block.element003 = "IO.Serial.Port" // Parâmetro 2
Block.element004 = 1
Block.element005 = "IO.Serial.BaudRate" // Parâmetro 3
Block.element006 = 19200
// Escreve o Bloco inteiro
Block.Write()
```

Ao usar o **Elipse E3**, a habilidade de criar *arrays* em tempo de execução permite o uso tanto de um Tag de Comunicação quanto de um Tag Bloco. Pode-se utilizar o método **Write** de um Driver para enviar os parâmetros diretamente para este Driver, sem a necessidade de criar um Tag, conforme o exemplo a seguir.

```
Dim arr(6)
' Configura os elementos do array
arr(1) = "IO.Type"
arr(2) = "Serial"
arr(3) = "IO.Serial.Port"
arr(4) = 1
arr(5) = "IO.Serial.BaudRate"
arr(6) = 19200
' Há dois métodos de enviar os parâmetros
' Método 1: Usando um Tag de Comunicação
tag.WriteEx arr
' Método 2: Sem utilizar um Tag
Driver.Write -1, 0, 0, 3, arr
```

Uma variação do exemplo anterior usa um *array* bidimensional.

```
Dim arr(10)
' Configura os elementos do array. Note que o array foi redimensionado
' para 10 elementos. Elementos vazios são ignorados pelo Driver
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
Driver.Write -1, 0, 0, 3, arr
```

Um Driver não valida nomes de parâmetros ou valores passados, por isto tenha cuidado ao escrever parâmetros e valores. O método **Write** falha se o *array* de configuração é criado incorretamente. Pode-se consultar o log de um Driver ou usar o parâmetro *writeStatus* do método **WriteEx** para descobrir a causa exata de um erro.

```
Dim arr(10), strError
arr(1) = Array("IO.Type", "Serial")
arr(2) = Array("IO.Serial.Port", 1)
arr(3) = Array("IO.Serial.BaudRate", 19200)
If Not Driver.WriteEx -1, 0, 0, 3, arr, , strError Then
    MsgBox "Falha ao configurar os parâmetros do Driver: " + strError
End If
```

IO.WorkOnline

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	4 (quatro)
Configuração por String	IO.WorkOnline

Este Tag informa o estado atual de um Driver e permite iniciar ou parar a camada física. Os valores possíveis são os seguintes:

- **0 - Driver Offline:** A camada física está fechada ou parada. Este modo permite uma configuração dinâmica dos parâmetros de um Driver através do Tag **IO.SetConfigurationParameters**
- **1 - Driver Online:** A camada física está aberta ou em execução. Enquanto está em modo **Online**, a camada física pode ser conectada ou desconectada e o estado atual pode ser conferido no Tag **IO.PhysicalLayerStatus**

No exemplo a seguir, utilizando o **Elipse E3**, um Driver é colocado em modo **Offline**, a porta COM é modificada e então é colocado em modo **Online** novamente.

```
'Configura o Driver em modo Offline
Driver.Write -1, 0, 0, 4, 0
'Muda a porta para COM2
Driver.Write -1, 0, 0, 3, Array("IO.Serial.Port", 2)
'Configura o Driver em modo Online
Driver.Write -1, 0, 0, 4, 1
```

O método **Write** pode falhar ao configurar um Driver em modo **Online**, ou seja, escrevendo o valor 1 (um). Neste caso, este Driver permanece em modo **Offline**. A causa da falha pode ser:

- Tipo de camada física configurada incorretamente, provavelmente um valor inválido foi configurado para a propriedade **IO.Type**
- Este Driver pode ter ficado sem memória
- A camada física pode ter deixado de criar a *thread* de trabalho. Procure no arquivo de log pela mensagem "Failed to create physical layer thread!"
- A camada física não conseguiu inicializar. A causa da falha depende do tipo de camada física. Pode ser um número de porta serial inválida, falha ao inicializar o Windows Sockets ou falha ao inicializar o TAPI (modem), entre outras. A causa é gravada no arquivo de log

IMPORTANTE

Mesmo que a configuração de um Driver para o modo **Online** seja bem-sucedida, isto não significa necessariamente que a camada física esteja pronta para uso, ou seja, pronta para executar operações de entrada e saída com um equipamento externo. O Tag **IO.PhysicalLayerStatus** deve ser verificado para assegurar que a camada física esteja conectada e preparada para a comunicação.

Propriedades

Estas são as propriedades gerais de todas as Interfaces de I/O suportadas.

IO.ConnectionMode

9 Controla o modo de gerenciamento da Conexão. Os valores possíveis são **0**: Modo automático, em que um Driver gerencia a conexão ou **1**: Modo manual, em que uma aplicação gerencia a conexão.

IO.GiveUpEnable

■ Quando configurada para Verdadeiro, define um número máximo de tentativas de reconexão. Se todas as reconexões falharem, um Driver entra em modo **Offline**. Se configurada para Falso, um Driver tenta até que uma reconexão seja bem-sucedida.

IO.GiveUpTries

9 Número de tentativas de reconexão antes que esta seja abortada. Por exemplo, se o valor desta propriedade é igual a 1 (um), um Driver tenta apenas uma reconexão quando a conexão é perdida. Se esta falhar, este Driver entra em modo **Offline**.

IO.InactivityEnable

☑ Configure em Verdadeiro para habilitar e em Falso para desabilitar a detecção de inatividade. A camada física é desconectada se está inativa por um certo período de tempo. A camada física é considerada inativa apenas se é capaz de enviar dados mas não de recebê-los de volta.

IO.InactivityPeriodSec

9 Número de segundos para a verificação de inatividade. Se a camada física está inativa por este período de tempo, então é desconectada.

IO.RecoverEnable

☑ Configure em Verdadeiro para habilitar um Driver a recuperar conexões perdidas e em Falso para deixar um Driver em modo **Offline** quando uma conexão é perdida.

IO.RecoverPeriodSec

9 Tempo de espera entre duas tentativas de conexão, em segundos.

NOTA

A primeira reconexão é executada imediatamente após a conexão ser perdida.

IO.StartOffline

☑ Configure em Verdadeiro para iniciar um Driver em modo **Offline** e em Falso para iniciar um Driver em modo **Online**.

NOTA

Não faz sentido modificar esta propriedade em tempo de execução, já que esta só pode ser modificada quando um Driver já está em modo **Offline**. Para configurar um Driver em modo **Online** em tempo de execução, escreva o valor 1 (um) no Tag **IO.WorkOnline**.

IO.TimeoutMs

9 Define o *time-out* da camada física, em milissegundos. Um segundo equivale a 1000 milissegundos.

IO.Type

A Define o tipo de interface física utilizada por um Driver. Os valores possíveis são os seguintes:

- **N ou None:** Não utiliza uma interface física, ou seja, um Driver deve fornecer uma interface personalizada
- **S ou Serial:** Utiliza uma porta serial local (COM n)
- **M ou Modem:** Utiliza um modem local, interno ou externo, acessado via TAPI (*Telephony Application Programming Interface*)

- **E ou Ethernet:** Utiliza um *socket* TCP/IP ou UDP/IP
- **R ou RAS:** Utiliza uma Interface **RAS** (*Remote Access Server*). Um Driver conecta-se a um equipamento RAS através da Interface **Ethernet** e então emite um comando **AT** (*dial*)

Configuração de Estatísticas

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** das estatísticas das Interfaces de Comunicação.

Tags de Comunicação

Tags de Estatísticas das Interfaces de Comunicação (N2/B2 = 0)

Os Tags descritos a seguir mostram estatísticas para todas as Interfaces de Comunicação.

IO.Stats.Partial.BytesRecv

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1101
Configuração por String	IO.Stats.Partial.BytesRecv

Este Tag retorna a quantidade de bytes recebidos na conexão atual.

IO.Stats.Partial.BytesSent

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1100
Configuração por String	IO.Stats.Partial.BytesSent

Este Tag retorna a quantidade de bytes enviados na conexão atual.

IO.Stats.Partial.TimeConnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1102
Configuração por String	IO.Stats.Partial.TimeConnectedSeconds

Este Tag retorna o número de segundos que um Driver está conectado na conexão atual ou 0 (zero) se um Driver está desconectado.

IO.Stats.Partial.TimeDisconnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1103
Configuração por String	IO.Stats.Partial.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que um Driver está desconectado desde o término da última conexão ou 0 (zero) se um Driver está conectado.

IO.Stats.Total.BytesRecv

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1001
Configuração por String	IO.Stats.Total.BytesRecv

Este Tag retorna a quantidade de bytes recebidos desde que um Driver foi carregado.

IO.Stats.Total.BytesSent

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1000
Configuração por String	IO.Stats.Total.BytesSent

Este Tag retorna a quantidade de bytes enviados desde que um Driver foi carregado.

IO.Stats.Total.ConnectionCount

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1004
Configuração por String	IO.Stats.Total.ConnectionCount

Este Tag retorna a quantidade de conexões que um Driver já estabeleceu, com sucesso, desde que foi carregado.

IO.Stats.Total.TimeConnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1002
Configuração por String	IO.Stats.Total.TimeConnectedSeconds

Este Tag retorna o número de segundos que um Driver permaneceu conectado desde que foi carregado.

IO.Stats.Total.TimeDisconnectedSeconds

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	0 (zero)
Parâmetro N4	1003
Configuração por String	IO.Stats.Total.TimeDisconnectedSeconds

Este Tag retorna o número de segundos que um Driver permaneceu desconectado desde que foi carregado.

Propriedades

Atualmente, não existem propriedades definidas especificamente para mostrar as estatísticas das Interfaces de Comunicação em tempo de execução.

Configuração da Interface Ethernet

Esta seção contém informações sobre a configuração dos **Tags de Comunicação** e das **Propriedades** da Interface **Ethernet**.

Tags de Comunicação

Tags da Interface Ethernet (N2/B2 = 4)

Os Tags descritos a seguir permitem controlar e identificar a Interface **Ethernet** em tempo de execução e também são válidos quando a Interface **RAS** está selecionada.

IMPORTANTE

Estes Tags estão disponíveis **SOMENTE** enquanto um Driver está em modo **Online**.

IO.Ethernet.IPSelect

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Leitura ou Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	0 (zero)
Configuração por String	IO.Ethernet.IPSelect

Indica o endereço IP ativo. Os valores possíveis são **0**: O endereço principal de IP está selecionado, **1**: O primeiro endereço IP alternativo ou de *backup* está selecionado, **2**: O segundo endereço IP alternativo ou de *backup* está selecionado ou **3**: O terceiro endereço IP alternativo ou de *backup* está selecionado.

Se a Interface **Ethernet** ou **RAS** está conectada, este Tag indica qual dos quatro endereços IP configurados está em uso. Se a Interface está desconectada, este Tag indica qual endereço IP é usado primeiro na próxima tentativa de conexão.

Durante o processo de conexão, se o endereço IP ativo não está disponível, a Interface de Comunicação tenta conectar-se usando o outro endereço IP. Se a conexão com o endereço IP alternativo funcionar, este é configurado como o endereço IP ativo (*switchover* automático).

Para forçar um *switchover* manual, escreva valores de 0 (zero) a três (3) neste Tag. Isto força a reconexão com o endereço IP especificado (**0**: Endereço principal, **1, 2, 3**: Endereços alternativos) se um Driver está atualmente conectado. Se um Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.IPSwitch

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Escrita
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	1 (um)
Configuração por String	IO.Ethernet.IPSwitch

Qualquer valor escrito neste Tag força um *switchover* manual. Se o endereço principal de IP está ativo, então o primeiro endereço IP alternativo ou de *backup* é ativado, e assim por diante passando por todos os endereços IP alternativos e voltando para o endereço principal até estabelecer uma conexão.

Se um Driver está desconectado, este Tag configura o endereço IP ativo para a próxima tentativa de conexão.

IO.Ethernet.SocketState

Tipo de Tag	Tag de Comunicação
Tipo de Acesso	Somente Leitura
Parâmetro N1	-1 (menos um)
Parâmetro N2	0 (zero)
Parâmetro N3	4 (quatro)
Parâmetro N4	2 (dois)
Configuração por String	IO.Ethernet.SocketState

A propriedade **Value** deste Tag corresponde a estados do *socket* em um mapa de bits:

- **Bit 0**: 0 (zero, não está em escuta) ou 1 (um, em escuta)

- **Bit 1:** 0 (zero, desconectado) ou 1 (um, conectado)

Propriedades

Estas propriedades controlam a configuração da Interface **Ethernet**.

NOTA

A Interface **Ethernet** também é usada pela Interface **RAS**.

IO.Ethernet.AcceptConnection

☑ Configure em Falso se um Driver não deve aceitar conexões externas, ou seja, se um Driver se comporta como mestre, ou configure em Verdadeiro para habilitar a recepção de conexões, ou seja, se um Driver se comporta como escravo.

IO.Ethernet.BackupEnable[2,3]

☑ Configure em Verdadeiro para habilitar o endereço IP alternativo ou de *backup*. Se a tentativa de reconectar com o endereço IP principal falhar, um Driver tenta utilizar um endereço IP alternativo ou de *backup*. Configure em Falso para desabilitar a utilização.

IO.Ethernet.BackupIP[2,3]

📌 Endereço IP alternativo ou de *backup* de um equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* de um equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.BackupLocalPort[2,3]

📌 Número da porta local a ser utilizada na conexão ao endereço IP alternativo ou de *backup* de um equipamento remoto. Usado apenas se a propriedade **IO.Ethernet.BackupLocalPortEnable** está configurada para Verdadeiro.

IO.Ethernet.BackupLocalPortEnable[2,3]

☑ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP alternativo ou de *backup* ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.BackupPort[2,3]

📌 Número da porta do endereço IP alternativo ou de *backup* de um equipamento remoto, usado juntamente com a propriedade **IO.Ethernet.BackupIP**.

IO.Ethernet.IPFilter

📌 Lista de endereços IPv4 ou IPv6 separados por vírgula, que define de quais endereços um Driver aceita ou bloqueia conexões. Pode-se utilizar asteriscos, como por exemplo "192.168.*.*", ou intervalos, como por exemplo "192.168.0.41-50", em qualquer parte dos endereços IP. Para bloquear um endereço IP ou um intervalos de endereços IP, use o caractere til ("~") no início do endereço, conforme os exemplos a seguir:

- **192.168.0.24:** Aceita apenas conexões do endereço IPv4 192.168.0.24
- **192.168.0.41-50:** Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.41 e 192.168.0.50
- **192.168.0.*:** Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255

- **fe80:3bf:877::*:*** (expande para **fe80:03bf:0877:0000:0000:0000:0000:0000**): Aceita conexões de endereços IPv6 no intervalo entre `fe80:03bf:0877:0000:0000:0000:0000:0000` e `fe80:03bf:0877:0000:0000:0000:0000:ffff:ffff`
- **192.168.0.10**, **192.168.0.15**, **192.168.0.20**: Aceita conexões dos endereços IPv4 192.168.0.10, 192.168.0.15 e 192.168.0.20
- **~192.168.0.95**, **192.168.0.***: Aceita conexões dos endereços IPv4 no intervalo entre 192.168.0.0 e 192.168.0.255, exceto o endereço IPv4 192.168.0.95

Quando um Driver recebe uma tentativa de conexão, a lista de filtros é percorrida sequencialmente da esquerda para a direita, procurando por uma autorização ou bloqueio específico para o endereço IP de onde veio a conexão. Se nenhum elemento da lista corresponde ao endereço IP, a autorização ou bloqueio são ditados pelo último elemento da lista:

- Se o último elemento da lista é uma autorização, como por exemplo "192.168.0.24", então todos os endereços IP não encontrados na lista são bloqueados
- Se o último elemento da lista é um bloqueio, como por exemplo "~192.168.0.24", então todos os endereços IP não encontrados na lista são autorizados

Se um endereço IP aparece em mais de um filtro da lista, o filtro mais à esquerda tem precedência. Por exemplo, no caso de "~192.168.0.95, 192.168.0.*", o endereço IP 192.168.0.95 se encaixa nas duas regras, mas a regra que vale é a mais à esquerda, "~192.168.0.95", e portanto o endereço IP é bloqueado.

Quando o **IOKit** bloqueia uma conexão, a mensagem "Blocked incoming socket connection from {IP}!" é logada.

No caso de conexões UDP em modo escuta em *broadcast*, em que um Driver pode receber pacotes de diferentes endereços IP, o bloqueio ou permissão é realizado a cada pacote recebido. Se um pacote é recebido de um endereço IP bloqueado, a mensagem "Blocked incoming packet from {IP} (discarding {N} bytes)!" é logada.

IO.Ethernet.ListenIP

A Endereço IP da interface local de rede por onde um Driver efetua e aceita conexões. Deixe esta propriedade vazia para efetuar e aceitar conexões por qualquer interface local de rede.

IO.Ethernet.ListenPort

9 Número da porta IP utilizada por um Driver para escutar conexões.

IO.Ethernet.MainIP

A Endereço IP de um equipamento remoto. Pode-se utilizar tanto o endereço numérico como o nome de *host* de um equipamento, como por exemplo "192.168.0.7" ou "SERVER2".

IO.Ethernet.MainLocalPort

9 Número da porta local a ser utilizada na conexão ao endereço IP principal de um equipamento remoto. Este valor é usado apenas se a propriedade **IO.Ethernet.MainLocalPortEnable** é igual a Verdadeiro.

IO.Ethernet.MainLocalPortEnable

☑ Configure em Verdadeiro para forçar o uso de uma porta local específica ao conectar ao endereço IP principal ou configure em Falso para utilizar qualquer porta local disponível.

IO.Ethernet.MainPort

9 Número da porta IP em um equipamento remoto, usado em conjunto com a propriedade **IO.Ethernet.MainIP**.

IO.Ethernet.PingEnable

■ Configure em Verdadeiro para habilitar o envio de um comando **ping** para o endereço IP de um equipamento remoto, antes de tentar conectar-se ao *socket*. O *time-out* de conexão do *socket* não pode ser controlado, por isto o envio de um comando **ping** antes de conectar-se é uma maneira rápida de detectar se a conexão vai falhar. Configure em Falso para desabilitar o comando **ping**.

IO.Ethernet.PingTimeoutMs

9 Tempo de espera por uma resposta de um comando **ping**, em milissegundos.

IO.Ethernet.PingTries

9 Número máximo de tentativas de comandos **ping**. O valor mínimo é 1 (um), incluindo o primeiro comando **ping**.

IO.Ethernet.ShareListenPort

■ Configure em Verdadeiro para compartilhar a porta de escuta com outros Drivers e processos ou Falso para abrir a porta de escuta em modo exclusivo. Para compartilhar uma porta de escuta com sucesso, todos os Drivers e processos envolvidos devem abrir esta porta em modo compartilhado. Quando uma porta de escuta é compartilhada, cada nova conexão é distribuída para um dos processos que estão escutando. Desta forma, se um Driver Escravo só suporta uma conexão por vez, pode-se utilizar várias instâncias deste Driver escutando na mesma porta, portanto simulando um Driver com suporte a múltiplas conexões.

IO.Ethernet.SupressEcho

■ Configure em Verdadeiro para eliminar o eco presente em uma comunicação. O eco é a recepção indesejada de uma cópia exata de todos os pacotes de dados que um Driver enviou para um equipamento.

IO.Ethernet.Transport

A Define o protocolo de transporte. Os valores possíveis são **T ou TCP**: Utiliza o protocolo TCP/IP ou **U ou UDP**: Utiliza o protocolo UDP/IP.

IO.Ethernet.UseIPv6

■ Configure em Verdadeiro para utilizar endereços IPv6 em todas as conexões Ethernet ou configure em Falso para utilizar endereços IPv4 (padrão).

Histórico de Revisões

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.0.7	02-09-2025	M. Ludwig	<ul style="list-style-type: none"> Driver atualizado para a biblioteca IOKit versão 3.0 e Visual Studio 2022 (<i>Case 38038</i>).
1.0.6	08/05/2017	F. Englert	<ul style="list-style-type: none"> Efetuosos ajustes neste Driver para utilizar a biblioteca IOKit versão 2.0.84 (<i>Case 22649</i>).

VERSÃO	DATA	AUTOR	COMENTÁRIOS
1.0.5	15/02/2017	M. Salvador	<ul style="list-style-type: none">• Criada uma opção de deslocamento (<i>offset</i>) manual, positivo ou negativo, para ser adicionado na estampa de tempo de eventos na aba Slave (<i>Case 21903</i>).
1.0.4	02/12/2016	M. Salvador	<ul style="list-style-type: none">• Implementado um atraso de 5 (cinco) segundos antes de solicitar uma Classe 0 (zero) após uma notificação de <i>reset</i> (<i>Case 21651</i>).
1.0.3	01/07/2015	M. Salvador	<ul style="list-style-type: none">• Implementadas melhorias no tratamento de desconexões para melhorar a notificação destas situações para um Mestre (<i>Case 18194</i>).
1.0.2	24/10/2014	M. Salvador	<ul style="list-style-type: none">• Versão inicial deste Driver.

Matriz

Rua Mostardeiro, 322/Cj. 902, 1001 e
1002

90430-000 — Porto Alegre — RS

Fone: (+55 51) 3346-4699

Fax: (+55 51) 3222-6226

E-mail: elipse-rs@elipse.com.br

Filial no Paraná

Av. Sete de Setembro, 4698/1708

80240-000 — Curitiba — PR

Fone: (+55 41) 4062-5824

E-mail: elipse-pr@elipse.com.br

Filial no Rio de Janeiro

Av. José Silva de A. Neto, 200/Bl. 4/Sl.
109B

22250-044 — Rio de Janeiro — RJ

Fone: (+55 21) 2430-5912

Suporte Técnico: (+55 21) 2430-5963

E-mail: elipse-rj@elipse.com.br

Filial em São Paulo

Rua dos Pinheiros, 870/Cj. 141 e 142

05422-001 — São Paulo — SP

Fone: (+55 11) 3061-2828

Fax: (+55 11) 3086-2338

E-mail: elipse-sp@elipse.com.br

Filial em Minas Gerais

Rua Antônio de Albuquerque, 156/705

30112-010 — Belo Horizonte — MG

Fone: (+55 31) 4062-5824

E-mail: elipse-mg@elipse.com.br

Filial em Taiwan

9F., No.12, Beiping 2nd St., Sanmin Dist.

807 — Kaohsiung City — Taiwan

Fone: (+886 7) 323-8468

Fax: (+886 7) 323-9656

E-mail: evan@elipse.com.br

Consulte nosso website para informações sobre o representante do seu estado.

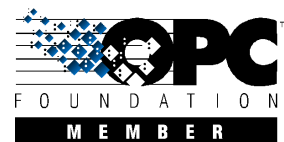
www.elipse.com.br

kb.elipse.com.br

forum.elipse.com.br

www.youtube.com/elipsesoftware

elipse@elipse.com.br



Gartner, Cool Vendors in Brazil 2014, April 2014.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability of fitness for a particular purpose.

Microsoft Partner

Gold Independent Software Vendor (ISV)